

---

# Laboratório de Introdução à Ciência da Computação I

## **Aula 3 – Vetores (continuação) e Matrizes**

Professores:

Vanderlei Bonato (responsável) - [vbonato@icmc.usp.br](mailto:vbonato@icmc.usp.br)

Luiz Henrique Kiehn (aluno PAE) - [lhkiehn@icmc.usp.br](mailto:lhkiehn@icmc.usp.br)

## Exercício: vetor

---

- Faça um programa que preencha um primeiro vetor com dez números inteiros e um segundo vetor com cinco números inteiros. O programa deverá mostrar uma lista dos números do primeiro vetor com seus respectivos divisores armazenados no segundo vetor, bem como suas posições
  - Para saber se um número inteiro é divisível por outro, deve-se calcular o resto da divisão
    - Exemplo em C

```
int x = 10, y = 2;
printf("%d\n",x%y);
```

# Exercício: vetor

---

- Faça um programa que realize as seguintes operações com a string “ano2010”:
  - Converta o “ano” para caracteres maiúsculos
    - Saída: ANO2010
  - Insira um caracter de espaço em branco na quarta posição sem que haja perda de conteúdo
    - Saída: ANO 2010
  - Inverta o ordem da string
    - Saída: 0102ONA

# Matrizes

---

- Modos de inicialização:

```
int mat[2][3];
```

```
int mat[2][3] = {{1,2,3},{4,5,6}};
```

```
int mat[][3] = {{1,2,3},{4,5,6}};
```

# Passagem de parâmetros

---

- Variável simples

```
main()
{
    int x = 5;
    func_a(x); //chamada da função
}
```

```
func_a(int y){ //função que recebe parâmetro
//func_a(y) int y;{ //outra forma de receber parâmetro
```

```
    ....
    ....
}
```

# Passagem de parâmetros

---

- Variável simples (mais de 1 parâmetro)

```
main()
{
  int x = 5;
  int y = 7;
```

```
  func_a(x, y); //chamada da função
}
```

```
func_a(int x, int y){ //função que recebe parâmetro
//func_a(x, y) int x, y;{ //outra forma de receber parâmetro
```

```
  ....
  ....
}
```

# Passagem de parâmetros

---

- Vetor

```
main()
{
    int x[3] = {1,2,3};
    func_a(x,3); //chamada da função
}
```

```
func_a(int y[],int tam){ //função que recebe parâmetro
    ....
    ....
}
```

# Passagem de parâmetros

---

- Matriz

```
main()
{
    int x[2][3] = {1,2,3};
    func_a(x,6); //chamada da função
}
```

```
func_a(int y[],int tam){ //função que recebe parâmetro
    ....
    ....
}
```

# Passagem de parâmetros

---

- Matriz (variação)

```
main()
{
    int x[2][3] = {1,2,3};
    func_a(x,2,3); //chamada da função
}
```

```
func_a(int y[][3], int l, int c)
{ //função que recebe parâmetro
    ....
    ....
}
```

# Exercício: matriz

---

- Escreva um programa que gere duas matrizes A e B quadradas de ordem 10 e que realize as seguintes operações:
  - transposta de A ( $A^t$ );
  - produto de  $AxB$ ;
  - calcule o determinante de A;

Implemente cada operação numa função independente, passando o conteúdo da matriz por parâmetro

Para facilitar a depuração do programa, imprima o conteúdo de A e B e o resultados das operações

# Apresentação do Projeto 1

---

- Projeto 1 – Editor de texto
- Data da apresentação e entrega: 10/05/2010 (durante a aula)
- Grupo: até 3 alunos
  
- Para implementar a janela do editor utilize a biblioteca Borland Graphics Interface (BGI) disponível em:  
<http://www.cs.colorado.edu/~main/bgi/dev-c++/>

# Teste da biblioteca BGI

---

```
#include <graphics.h>

int main( ){
    char vetor[200] = "";
    initwindow(640, 480, "First Sample");
    //circle(100, 50, 40);
    moveto (0, 0);
    outtext (vetor);
    char c[1];
    int x = 0;
    while (c[0]!='q') {
        c[0] = getch();
        outtext (c);
    }
}
```

# Referências

---

Ascencio AFG, Campos EAV. Fundamentos de programação de computadores. São Paulo : Pearson Prentice Hall, 2006. 385 p.

VICTORINE VIVIANE MIZRAHI, Treinamento em Linguagem C – Módulo 1 e Módulo 2, Makron Books, 1990.

<http://www.cs.colorado.edu/~main/bgi/dev-c++/>